Augmented Reality Interface for Constrained Learning from Demonstration

Matthew B. Luebbers, Connor Brooks, Minjae John Kim, Daniel Szafir, Bradley Hayes
University of Colorado Boulder
Boulder, CO, USA
{matthew.luebbers, connor.brooks, minjae.kim, daniel.szafir, bradley.hayes}@colorado.edu

***Abstract*—This paper presents a novel augmented reality (AR) interface for the visualization and directed control of robot skill Learning from Demonstration (LfD). This system is designed for use with the Concept-Constrained Learning from Demonstration (CC-LfD) algorithm for general robotic arm manipulation tasks, in which trajectories in an LfD system are subjected to various constraints during the learning process to ensure that the desired skill is learned properly. This system provides an interactive visualization of observed and learned robot trajectories, as well as the active predicate constraints applied for CC-LfD. In this paper, we describe current work on a system for helping users give improved trajectory demonstrations using AR visualizations of constraints, as well as a planned human subjects experiment to evaluate the usefulness of this system. Additionally, we discuss future extensions of this work involving using an AR interface to modify existing trajectory demonstrations.**

***Index Terms*—Learning from Demonstration, Augmented Reality, Human-Robot Interaction**

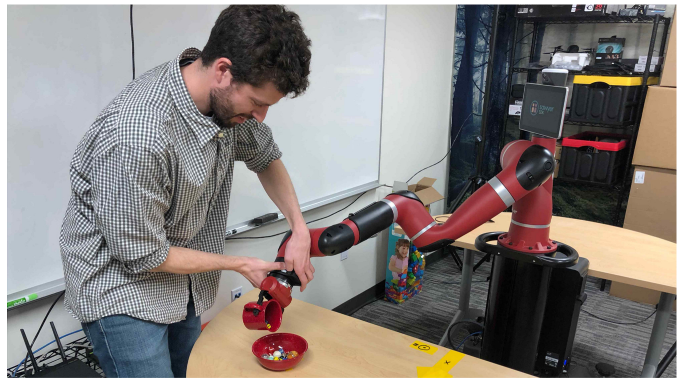

Fig. 1. Sawyer robot manufacturing arm being guided through a kinesthetic demonstration of a sample cup-pouring manipulation task for use in CC-LfD. Credit Mueller et al. [10]

I. Introduction

Robot skill Learning from Demonstration (LfD) is a broad collection of techniques that involve robots learning skills and behaviors from a set of 'ground truth' human demonstrations [2] [12] [13]. These demonstrations typically take the form of guiding the robot through the desired task, from an initial state to a goal state, either physically, in the case of kinesthetic LfD, via teleoperation, or in simulation. Alternative terms for LfD common in robotics literature include *programming by demonstration* [3], *apprenticeship learning* [1], and *human-agent transfer* [14].

LfD is an alternative to manual specification of the skill to be learned, either via explicit trajectory programming or manual tuning of an objective function. This allows robots to learn tasks without requiring expert knowledge of programming and the specific robotic system, thus allowing for lay-users to teach a robot successfully. Despite these prominent advantages, LfD is prone to a few different classes of issues related to practical use. First, the technique tends to suffer from a lack of robustness to alterations in start conditions i.e., it suffers from overfitting. This problem is hardly unique to LfD among robotic control techniques, whether they are learning based or purely analytical. Stemming from this overfitting also comes an oversensitivity to side-effects and unintended behaviors deleterious to the task being learned, especially when poor quality demonstrations are included.

To combat the negative effects of non-perfect demonstrations, Mueller et al. [10] devised an algorithm known as Concept-Constrained Learning from Demonstration (CC-LfD). The CC-LfD algorithm proceeds thusly:

- Multiple trajectories are gathered from human kinesthetic demonstrations, with annotated human-defined predicate constraints present along all or part of the trajectories.
- The trajectories are sequenced and grouped into keyframes after undergoing Dynamic Time Warping (DTW) [15] to align the time-series data of demonstrations given at differing speeds.
- Keyframes for the learned skill are modeled generatively with Gaussian Kernel Density Estimation [4] from the demonstrated training data.
- Final trajectories are created from sampled points within the learned keyframes, while rejecting any such points that violate any of the applied constraints.

CC-LfD demonstrates significant improvement in learned skill accuracy and convergence over standard kinesthetic LfD. Assuming appropriate constraints are selected, the algorithm guards against errors in demonstration and provides an additional layer of concept modeling on-top of what is typically a fairly rigid set of motion primitives.

However, as was stated prior, one of the goals of LfD as a technique is to allow lay-users to easily and effectively program a robot without requiring detailed knowledge of programming or the robot system in question. To this ideal, the end-to-end CC-LfD system is currently underpowered. Concept constraints take the form of pre-programmed predicates involving some subset of the robot's state space. These constraints are defined beforehand programmatically, and then are applied to a trajectory live as it is being demonstrated. Since there is no visualization of these constraints, the process of applying them while simultaneously providing a demonstration is unintuitive and difficult. What's more, validating constraints and learned trajectories is arduous and cannot be accomplished live, or within the context of the robot's environment.

To solve these issues, we are developing a novel augmented reality (AR) interface for conducting CC-LfD. This interface provides a visualization of demonstrated or learned trajectories, along with their constraints, overlaid onto the real world, allowing a user to easily contextualize the constraints and visualize if they are broken or satisfied in real space.

The full AR for CC-LfD system will also include the abilities for a user to edit constraints, edit where constraints are and are not applied, edit keyframes, and define brand new constraints from a set of parameterizable presets. The goal of our integration of AR visualization into the CC-LfD system is to make CC-LfD user-friendly and powerful by helping users give properly constrained demonstrations to a robot.

II. Related Work

Commercial augmented reality headsets for development have recently become available for widespread research use, with the Microsoft HoloLens and Meta 2 headsets releasing to developers in 2016, and the Magic Leap One releasing in 2018. In that time since these headsets have been made available, there has been much research conducted into AR applications for robotics. One theme that has been researched in this space with particular relevance to our AR for CC-LfD system is the visualization of robot trajectories or motion plans. This concept has been applied both to manufacturing robots [11] and flying mobile robots [16]. AR has also successfully been utilized to create user-interfaces to aid in the explicit programming of robot trajectories by projecting constraint data onto the real world space [5] and to display internal robot state for understanding and debugging of a multi-robot system [6].

Despite this widespread recent adoption of AR for human-robot interaction (HRI) and robot teleoperation, there has to this point been a general absence of research on AR interfaces for Learning from Demonstration. There are a couple of systems which utilize AR headsets as a tool in LfD or LfD-like contexts, but they differ substantially in their approach and purpose. Li et al. [8] implemented a system that allows a user to guide a haptic controller to perform teleoperated programming by demonstration while wearing an AR headset that provides a visualization of environmental obstacles to be avoided. Meanwhile, Liu et al. [9] propose a system to use an AR headset as a data-overlay, displaying relevant high-level state and action information in an interpretable, interactive data structure known as a Temporal And-Or graph (T-AOG).

No system currently exists that fills the role intended by the AR interface described in this paper. Our goal in developing AR for CC-LfD is to provide an intuitive in-place visualization of the robot's internal state and learning process in addition to serving as a tool to positively augment that learning process. We believe AR is uniquely suited to improve LfD by both helping a user understand what trajectories are needed to help the robot learn the skill properly and displaying this information directly in the physical space occupied by the human and robot, a factor which has been shown to lead to safer human control of robots [7].

III. Initial Integration of AR for CC-LFD

In this section, we describe our initial implementation and a proposed experiment for integrating AR into CC-LFD.

A. Implementation

Our initial implementation is designed in order to facilitate improved LFD trajectory demonstrations with predefined annotated constraints. To do this, we propose to create AR visualizations of applied constraints that are projected onto the space in which a user gives a sample trajectory. These constraints will initially take two forms: height constraints and rotational constraints.

Height constraints correspond to a limit on the height that the robot should reach while performing a task. This could be important in tasks such as a robot transporting a fragile object that would break if dropped far from the ground. Rotation constraints correspond to a limit on the rotation of the robot away from a given reference point with respect to two of the axes of rotation. This constraint can be used in situations such as a robot transporting a cup of coffee. In this case, the degrees of pitch and roll of the cup should be limited, but yaw is permitted. Sample visualizations of these constraints applied to recorded robot trajectories are shown in Fig. 2 and Fig. 3.

These constraints are applied to the position and orientation of the robot's end effector as a demonstration is given. An applied height constraint will appear as a virtual plane

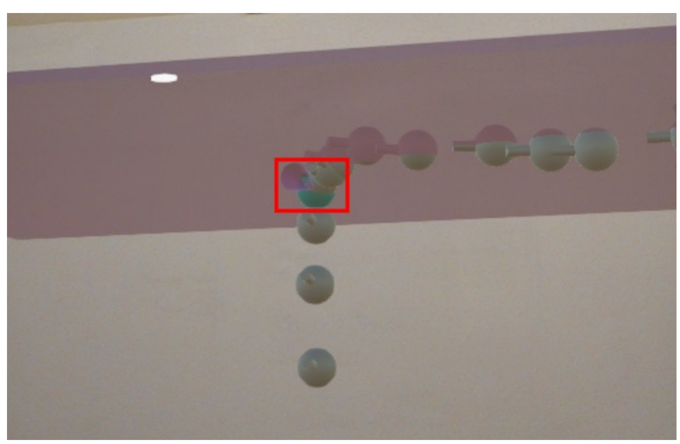

Fig. 2. Visualization of a height constraint, along with a rotational constraint, active at a keyframe (inside the red box). The height constraint is displayed as a purple plane, and the rotational constraint is displayed as a purple cone. The selected trajectory keyframe node is highlighted in green, indicating no violations.

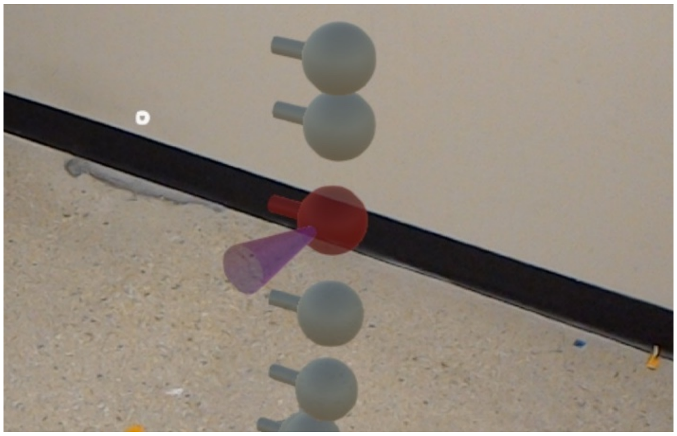

Fig. 3. Visualization of a rotational constraint active at a keyframe, displayed as a purple cone. The selected trajectory keyframe node is highlighted in red, indicating a rotational violation. This is consistent with the visualization where the node's leg points outside of the cone.

rendered to demarcate the out-of-bounds barrier above which the constraint will be violated. (Fig. 2) An applied rotational constraint will appear as a cone around the end effector's current position, with rotation outside of the allowed bounds for the two relevant axes causing the end effector to point outside of the cone. (Fig. 3)

These constraints will appear only when users mark them active (e.g., the robot only needs to worry about holding the cup of coffee upright after the segment of the skill in which it picks up the cup). Users will be able to toggle predefined constraints as active or inactive through using a button on the robot's arm. If the state of the end effector violates any active constraints, the violated constraint(s) will be highlighted in red and additional visualization will notify the user that there are currently violated constraints.

B. Experiment

In order to evaluate the usefulness of our system, we propose the following experiment.

1) Experimental Scenario: We will have a series of trajectories without constraint annotation, along with a set of pre-defined constraints, loaded onto a Sawyer robot arm for a complex manipulation task in which precise selection and placement of constraints is necessary to ensure completion (e.g., retrieving an object in a highly cluttered environment, where the arm must not collide with a number of obstacles). The pre-loaded trajectories, though individually successful, will generate invalid trajectories when combined in an LfD model due to their high variance, leading to task failure. The participant's goal will be to successfully demonstrate a single constraint-annotated trajectory for the task using the given constraints, so that the pre-loaded trajectories are repaired and the task completes successfully after running CC-LfD.

After the participants are briefed on the intended task, the constraints available to them, and how to perform a kinesthetic demonstration, they will proceed to deliver constraint-annotated demonstrations under three experimental conditions presented in a random order, differing in the interface for visualizing constraints made available to them. Following the completion of all three trials, they will be presented with a second, similar manipulation task to demonstrate a constraint-annotated trajectory, for which they will be provided with an interface of their own choosing from the previous conditions. Each interface condition is described below.

2) Condition 1: No Visualization: In this condition, the participant will have only the verbal description of each of the pre-built constraints available to them. They will then proceed as normal in standard CC-LfD, pressing buttons on the robot arm's cuff to start and stop the application of individual constraints. This condition is analogous to the current state of the art interface for CC-LfD.

3) Condition 2: 2D Visualization: In this condition, the participant will be given a visualized representation of each of the pre-built constraints overlaid on a simulated robot arm on a 2-dimensional screen (a monitor or tablet) that they will have constant access to throughout the condition. They will apply constraints as normal in CC-LfD.

4) Condition 3: AR Visualization: In this condition, the participant will have access to the AR for CC-LfD interface. They will wear a Microsoft HoloLens headset, allowing for visualization of constraints holographically overlaid on the robot arm. They will apply constraints as normal in CC-LfD.

5) Evaluation and Hypothesis: We propose to evaluate the interfaces on two metrics - their ability to induce proper concept-annotated demonstrations that successfully complete the task, and their perceived usability and user-friendliness. To accomplish this, we will structure the evaluation as a hybrid between-subjects/within-subjects study, where each participant completes each experimental condition in a random order, but where objective data on task completion accuracy is only considered from the first condition seen, in order to guard against the significant carryover effects stemming from repeated performance of the task.

In addition to the task accuracy data collected, a series of short surveys following each trial and a longer exit survey involving comparison of the conditions will provide information about the perceived usability of each interface. Recording which interface participants choose to use for the second, similar task will provide a more concrete indication of which interface is preferred.

Our hypothesis is that Condition 3 will have the highest scores both for accuracy and perceived usability, followed by Condition 2 for both metrics, followed by condition 1 for both metrics. This would demonstrate not only that the AR interface for CC-LfD is more usable for a lay-user than existing interfaces, but also that an augmented reality interface provides additional benefits to understanding and algorithm success on top of visualization with a 2-dimensional screen.

IV. Further Extensions

In further extensions of this work, we are developing the ability for users to modify existing trajectories and constraints using our AR visualizations. The in-place context provided through the use of AR would prove highly useful in this regard. First, we will allow users to edit where constraints are and are not active on a constraint-annotated trajectory within the AR interface. This can be accomplished by giving the user the ability to indicate starting and ending keyframes for any constraints loaded in for the current learning instance. The user will also be able to parametrically edit any of the constraints loaded in, as well as define new constraints via a set of constraint predicates and add these to an existing trajectory.

Second, users will also be capable of editing trajectories themselves, demonstrated or learned. This will be done through users selecting virtual spheres marking keyframes of an existing trajectory, then dragging and rotating them according to the new desired position and rotation. Other nearby keyframes in the trajectory will be smoothed with the new keyframe position. Users will also be able to visualize constraints and constraint violations during this step if desired.

These two tools will allow users to correct and modify existing trajectories while visualizing desired constraints in order to ensure that the robot learns the skill in the manner desired, thus adding an additional layer of utility to the interface.

V. Conclusion

In this paper, we presented our work on a novel interface for conducting Concept-Constrained Learning from Demonstration (CC-LfD) utilizing the Microsoft HoloLens augmented reality (AR) headset. We described a proposed human user study to evaluate the success of this interface over alternative interfaces for the CC-LfD algorithm. We additionally described proposed additions to the end-to-end AR system to achieve greater usability, verifiability, flexibility, and capability. If successful, this interface will demonstrate the value of AR interfaces for the broad techniques encompassed by robot skill Learning from Demonstration, as well as for human-robot interaction as a whole.

References

[1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning (ICML 2004)*, page 1. ACM, 2004.

[2] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57 (5):469–483, 2009.

[3] Aude Billard, Sylvain Calinon, Ruediger Dillmann, and Stefan Schaal. Robot programming by demonstration. In *Springer Handbook of Robotics*, pages 1371–1394. Springer, 2008.

[4] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification*. John Wiley & Sons, 2012.

[5] HC Fang, SK Ong, and AYC Nee. Interactive robot trajectory planning and simulation using augmented reality. *Robotics and Computer-Integrated Manufacturing*, 28(2):227–237, 2012.

[6] Fabrizio Ghiringhelli, Jérôme Guzzi, Gianni A Di Caro, Vincenzo Caglioti, Luca M Gambardella, and Alessandro Giusti. Interactive augmented reality for understanding and analyzing multi-robot systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, pages 1195–1201. IEEE, 2014.

[7] Hooman Hedayati, Michael Walker, and Daniel Szafir. Improving collocated robot teleoperation with augmented reality. In *ACM/IEEE International Conference on Human-Robot Interaction (HRI 2018)*, pages 78–86. ACM, 2018.

[8] Pai-Chia Li, Chih-Hsing Chu, et al. Augmented reality based robot path planning for programming by demonstration. In *A-DEWS 2016-Innovation of Life in Asia-Asian Design Engineering Workshop*, pages 127–131, 2016.

[9] Hangxin Liu, Yaofang Zhang, Wenwen Si, Xu Xie, Yixin Zhu, and Song-Chun Zhu. Interactive robot knowledge patching using augmented reality. In *IEEE International Conference on Robotics and Automation (ICRA 2018)*, pages 1947–1954. IEEE, 2018.

[10] Carl Mueller, Jeff Venicx, and Bradley Hayes. Robust robot learning from demonstration and skill repair using conceptual constraints. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2018)*, 2018.

[11] Eric Rosen, David Whitney, Elizabeth Phillips, Gary Chien, James Tompkin, George Konidaris, and Stefanie Tellex. Communicating robot arm motion intent through mixed reality head-mounted displays. *arXiv preprint arXiv:1708.03655*, 2017.

[12] Leonel Rozo, Pablo Jiménez, and Carme Torras. A robot learning from demonstration framework to perform force-based manipulation tasks. *Intelligent Service Robotics*, 6(1):33–51, 2013.

[13] Stefan Schaal. Learning from demonstration. In *Advances in Neural Information Processing Systems*, pages 1040–1046, 1997.

[14] Matthew E Taylor, Halit Bener Suay, and Sonia Chernova. Integrating reinforcement learning with human demonstrations of varying ability. In *International Conference on Autonomous Agents and Multiagent Systems (ICAAMS 2011)*, pages 617–624. International Foundation for Autonomous Agents and Multiagent Systems, 2011.

[15] Najdan Vuković, Marko Mitić, and Zoran Miljković. Trajectory learning and reproduction for differential drive mobile robots based on gmm/hmm and dynamic time warping using learning from demonstration framework. *Engineering Applications of Artificial Intelligence*, 45: 388–404, 2015.

[16] Michael Walker, Hooman Hedayati, Jennifer Lee, and Daniel Szafir. Communicating robot motion intent with augmented reality. In *ACM/IEEE International Conference on Human-Robot Interaction (HRI 2018)*, pages 316–324. ACM, 2018.